



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/665,564

09/22/2003

Louis R. Degenaro

YOR920030126US1

6151

48150

7590

03/22/2006

MCGINN INTELLECTUAL PROPERTY LAW GROUP, PLLC  
8321 OLD COURTHOUSE ROAD  
SUITE 200  
VIENNA, VA 22182-3817

EXAMINER

SYED, FARHAN M

ART UNIT

PAPER NUMBER

2165

DATE MAILED: 03/22/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

10/665,564

Applicant(s)

DEGENARO ET AL.

Examiner

Farhan M. Syed

Art Unit

2165

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 22 September 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-37 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-37 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 22 September 2003 is/are: a) ☐ accepted or b) ☒ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |   |   |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)   | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)  | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date <u>20031230/ 20040113</u> . | 6) <input type="checkbox"/> Other: _____  |

**DETAILED ACTION**

1. Claims 1-37 are pending.

***Drawings***

2. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description: Figure 4, item 420; Figure 8, item 840. Corrected drawing sheets in compliance with 37 CFR 1.121(d), or amendment to the specification to add the reference character(s) in the description in compliance with 37 CFR 1.121(b) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

***Specification***

3. The specification is objected to because of the following informalities: On page 40, line 13 (i.e. line 13 of page 40), paper is not a suitable machine-readable data storage media. Appropriate correction is required.

***Claim Objections***

4. Claims 6 and 10 are objected to because of the following informalities: On line 10 of claims 6 and 10 (i.e. line 10, claim 6 and 10), after the word 'time', there should be a '.' as oppose to a ';'. Appropriate correction is required.

***Claim Rejections - 35 USC § 112***

5. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

6. Claim 33 is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

As per claims 33, it clearly recites a "method of refactoring actual resources without alteration." The specifications, submitted by the applicant, does not detail how an ordinary person skilled in the art may refactor actual resources without alteration.

7. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

8. Claims 1, 4, 8, 14, 17, 21 and 31-33 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

The term "refactoring" in claims 1, 14, and 31-33 is a relative term, which renders the claim indefinite. The term "refactoring" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention. The examiner will assume that the term means replicating or duplicating.

The term "relationship carrying semantic" in claim 4 and 17 is a relative term, which renders the claim indefinite. The term "relationship carrying semantic" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention. The examiner will assume that the term means an identifier or unique ID.

The term "model flattening relationship with a semantic meaning of reachability" in claim 8 and 21 is a relative term which renders the claim indefinite. The term "model

Art Unit: 2165

flattening relationship with a semantic meaning of reachability" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention. The examiner will assume that the term means a relational database.

***Claim Rejections - 35 USC § 101***

9. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

10. Claim 32 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

As per claim 32, it clearly recites a "signal-bearing medium." Signal-bearing medium is not tangible, and cannot tangibly embody a computer program or process since a computer cannot understand/realize (i.e. execute) the computer program or process when embodied on the data signal. Computer program or processes are only realized within the computer when stored in a memory or storage element (such as RAM or ROM). Therefore, a data signal does not meet the "useful, concrete, and tangible" requirement as set forth in *State Street*, 149 F.3d at 1373, 47 USPQ2d at 1601-02, and hence claims 25-32 are non statutory under 35 U.S.C. 101.

Art Unit: 2165

11. Claim 33 is rejected under 35 U.S.C. 101 because the disclosed invention is inoperative and therefore lacks utility.

As per claims 33, it clearly recites a "method of refactoring actual resources without alteration." The specifications, submitted by the applicant, does not detail how an ordinary person skilled in the art may refactor actual resources without alteration.

***Claim Rejections - 35 USC § 102***

12. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

13. Claims 1-26, 31-35 are rejected under 35 U.S.C. 102(a) as being anticipated by a non-patent literature entitled "XTABLES: Bridging relational technology and XML" by J.E. Funderburk, G. Kiernan, J. Shanmugasundaram, E. Shekita, and C. Wei, pages 616-641 (known hereinafter as Funderburk).

As per claims 1, 14, 31, and 32, Funderburk teaches a method of refactoring a plurality of actual resources without alteration into a collection of virtual resources customized to a particular audience, said method comprising (i.e. *"One of the features provided by XTABLES is the ability to create XML views of existing relational data."* *"Users can then create application-specific XML views on top of the default XML view."* The preceding text clearly indicates that a collection of virtual resources is a specific XML view and existing relational data is the actual resource.)(Page 616, paragraph 2): constructing at least one virtual resource (i.e. *"Figure*

*2 illustrates the default view for a simple purchase-order database. The database consists of three tables, one to track customer orders, a second to track items associated with an order, and a third to track the payments due for each order. Items and payments are related to orders by an order identifier (oid). In the default XML view, top-level elements correspond to tables, with table names appearing as tags. Row elements are nested under these. Within a row element, column names appear as tags and column values appear as text. Although not shown, an XML schema associated with the default view captures primary- and foreign-key relationships.*" The preceding text clearly indicates that at least one virtual resource is the default XML view.)(Page 620); connecting at least one actual resource to the at

least one virtual resource (i.e. *"Figure 2 illustrates the default view for a simple purchase-order database. The database consists of three tables, one to track customer orders, a second to track items associated with an order, and a third to track the payments due for each order. Items and payments are related to orders by an order identifier (oid). In the default XML view, top-level elements correspond to tables, with table names appearing as tags. Row elements are nested under these. Within a row element, column names appear as tags and column values appear as text. Although not shown, an XML schema associated with the default view captures primary- and foreign-key relationships."* The preceding text clearly indicates that connecting is corresponding, at least one actual resource is the database consisting of three tables, and virtual resource is the default XML view.)(Page 620); retrieving the at least one virtual resource (i.e. *"Figure 2 illustrates the default view for a simple purchase-order database. The database consists of three tables, one to track customer orders, a second to track items associated with an order, and a third to track the payments due for each order. Items and payments are related to orders by an order identifier (oid). In the default XML view, top-level elements correspond to tables, with table names appearing as tags. Row elements are nested under these. Within a row element, column names appear as tags and column values appear as text. Although not shown, an XML schema associated with the default view captures primary- and foreign-key relationships."* The preceding text clearly indicates that retrieving is displaying the default XML view)(Page 620); and extracting at least one descriptor from said at least one retrieved virtual resource (i.e. *"Figure 2 illustrates the default view for a*



*simple purchase-order database. The database consists of three tables, one to track customer orders, a second to track items associated with an order, and a third to track the payments due for each order. Items and payments are related to orders by an order identifier (oid). In the default XML view, top-level elements correspond to tables, with table names appearing as tags. Row elements are nested under these. Within a row element, column names appear as tags and column values appear as text. Although not shown, an XML schema associated with the default view captures primary- and foreign-key relationships."* The preceding text clearly indicates that at least one descriptor is a tag.) (Page 620).

As per claims 2 and 15, Funderburk teaches a method wherein said connecting comprises directly mapping the at least one actual resource to the at least one virtual resource (i.e. *"XTABLES does this by automatically mapping the schema and data of the underlying relational database system to a low-level default XML view."* The preceding text clearly indicates that at least one actual resource is the underlying relational database system and at least one virtual resource is the low-level default XML view.)(Page 616).

As per claims 3 and 16, Funderburk teaches a method wherein the constructing comprises at least one of: renaming a method; hiding a method; composing a method; renaming an attribute; hiding an attribute; composing an attribute; assigning to at least one domain; designating as a collection; assigning to at least one validator; assigning a description; designating as at least one of ready and not ready; and assigning a last modified date and time (i.e. *"A query is initially parsed and converted from XQuery to an intermediate query representation called the XML Query Graph Model (XQGM). The query is then composed with the XML views it references, and rewrite optimizations are performed to eliminate the construction of intermediate XML fragments, unroll recursion, and push down predicates."* The preceding text clearly indicates that renaming a method, hiding a method, etc. is example of XML fragments)(Page 622).

As per claims 4, 8, 17 and 21, Funderburk teaches a method wherein virtual resources are connected to each other through a relationship carrying semantic that can be leveraged by a consumer of resources, comprising (i.e. *"Users can then create application-specific XML views on top of the default XML view. These application-specific views are created using XQuery, a general-purpose, declarative XML query language currently being standardized by the W3C (World Wide Web Consortium)."* The preceding text clearly indicates that a relationship carrying semantic is the creation of application-specific XML views on top of the default XML view, where XML views are virtual resources and consumer of resources are users.)(Page 616): constructing at least one virtual relationship between at least two virtual resources (i.e. *"Continuing the example, suppose that a user wants to publish the purchase-order database as a list of orders in the XML format shown in Figure 3. There, each order appears as a top-level element, with its associated items and payments (ordered by due date) nested under it. To transform the default view into the desired XML format, a user-defined view called "orders" is created, as shown in Figure 4. The view definition is fairly straightforward. An XQuery FLWR (for, let, where, return) expression (lines 2–22) is used to construct each order element. The "for" clause on line 2 causes the variable \$order to be bound to each "row" element of the order table. The XPath expression appearing in line 2 describes how to extract each "row" element from the order table: start at the root of the default view, navigate to each "order" element nested under it, and then navigate to each "row" element nested under those "order" elements. The constructor for each new "order" element is given in lines 4–22. For a given order, nested FLWR expressions are used to construct its list of associated items (lines 6–13) and payments (lines 14–21). The predicate on line 8 (\$order/id\_ \$item/oid) is used to join an order with its items. Similarly, the predicate on line 16 (\$order/id\_ \$payment/oid) is used to join an order with its payments."* The preceding text clearly indicates that at least one virtual relationship is orders and the two virtual resources are the default XML view and the user-defined view.)(Page 620-621); coupling at least one actual relationship implementor

to at least one virtual relationship (i.e. Per the preceding text, the orders, which is a virtual relationship is based on one of the three tables, where an actual relationship is the customer order.)(Page 620); performing at least one retrieval of a virtual relationship (i.e. *"An XQuery FLWR (for, let, where, return) expression (lines 2–22) is used to construct each order element."* The preceding text illustrates a retrieval of orders from the purchase-order database to the user-defined XML view 'Order.')(Page 620); and extracting at least one descriptor from at least one retrieved virtual relationship (i.e. Per the preceding text, the XML tags, which are descriptors, are embedded in an XML view.)(Page 620).

As per claims 5, 9, 18, and 22, Funderburk teaches a method wherein said coupling comprises: directly mapping said at least one actual relationship implementor to said at least one virtual relationship (i.e. *"As shown, each edge is uniquely identified by the identifier fields of the source and destination nodes (the sid and did fields). Each edge also contains the name, value, and type information about its destination node. The order among sibling subelements is captured using the ordinal field. In our example, the edge pointing to the root XML element ("PurchaseOrder") is mapped to the first row. Its sid field is 0, which represents the identifier of the document root. The edges pointing to the BuyerName and Date attributes of the "PurchaseOrder" element are mapped to the second and third row, respectively. Note that these are related to the purchase order using the sid field. Similarly, the "ItemsBought" and "Payments" subelements of a "PurchaseOrder" element are represented by the fourth and fifth row, respectively. The ordinal field captures their relative order. The other edges of the document are stored similarly."* The preceding text clearly indicates that the virtual relationship is the purchase order and the actual relationship implementor is the first row of a table.)(Page 636).

As per claims 6, 10 19, and 23, Funderburk teaches a method wherein the relationship constructing comprises at least one of: assigning a root virtual resource name; assigning a target virtual resource name; assigning a relationship name; assigning a relationship type; assigning a description; assigning a target instance naming scheme; designating as at least one of ready and not ready; and assigning a last modified date and time (i.e. *"As shown, each edge is uniquely identified by the identifier fields of the source and destination nodes (the sid and did fields). Each edge also contains the name, value, and type information about its destination node. The order among sibling subelements is captured using the ordinal field. In our example, the edge pointing to the root XML element ("PurchaseOrder") is mapped to the first row. Its sid field is 0, which represents the identifier of the document root. The edges pointing to the BuyerName and Date attributes of the "PurchaseOrder" element are mapped to the second and third row, respectively. Note that these are related to the purchase order using the sid field. Similarly, the "ItemsBought" and "Payments" subelements of a "PurchaseOrder" element are represented by the fourth and fifth row, respectively. The ordinal field captures their relative order. The other edges of the document are stored similarly."* The preceding text clearly indicates that the 'PurchaseOrder' is assigning a relationship name between a virtual resource, which is the XML view, and the actual resource, which is a table within a relational database.)(Page 636).

As per claims 7,12, 20 and 25, Funderburk teaches a method wherein the retrieving comprises locating virtual relationships by at least one of: a domain; a name; a root; a type; and a target (i.e. *"As shown, each edge is uniquely identified by the identifier fields of the source and destination nodes (the sid and did fields). Each edge also contains the name, value, and type information about its destination node. The order among sibling subelements is captured using the ordinal field. In our example, the edge pointing to the root XML element ("PurchaseOrder") is mapped to the first row. Its sid field is 0, which represents the identifier of the document root. The edges pointing to*

*the BuyerName and Date attributes of the "PurchaseOrder" element are mapped to the second and third row, respectively. Note that these are related to the purchase order using the sid field. Similarly, the "ItemsBought" and "Payments" subelements of a "PurchaseOrder" element are represented by the fourth and fifth row, respectively. The ordinal field captures their relative order. The other edges of the document are stored similarly."* The preceding text clearly indicates that the 'PurchaseOrder' is an example of a name.)(Page 636).

As per claims 11 and 24, Funderburk teaches a method wherein the retrieving comprises locating virtual resources by at least one of (i.e. *"Continuing the example, suppose that a user wants to publish the purchase-order database as a list of orders in the XML format shown in Figure 3."* The preceding text clearly indicates that retrieving is publishing.)(Page 620): a domain; a name; and a relationship. (i.e. *"To transform the default view into the desired XML format, a user-defined view called "orders" is created, as shown in Figure 4."* The preceding text clearly indicates that a user-defined view, which is an example of a virtual resource, called "orders" is, which is an example of name.)(Page 620).

As per claims 13 and 26, Funderburk teaches a method, wherein the descriptor validator information is employed to limit actual resource usage (i.e. *"Another feature provided by XTABLES is the ability to query XML views of relational data. This is important because users often need only a subset of a view's data. Moreover, users often need to synthesize and extract data from multiple views. In XTABLES, queries are specified using XQuery, the same language used to specify XML views. XTABLES executes queries efficiently by performing XML view composition so that only the desired relational data items are materialized."* The preceding text clearly indicates that a descriptor validator information is an XML tag that is used to generate XML views based on the relational data to synthesize and extract data from multiple views, thus limiting the actual resource, which is the relational data, usage.) (Pages 616-617).

As per claim 27, Funderburk teaches a system comprised of a plurality of actual resources, a service to manage descriptions of said actual resources comprising: defining at least one virtual domain to satisfy a requirement analysis (i.e. *"Users can query XML documents using the same query language that they use to create and query XML views of relational data. In addition, users can issue queries that span XML documents and XML views of relational data. As a result, users are provided with unified access to both relational data and XML documents, without having to deal with separate databases."* The preceding text clearly indicates that a requirement analysis is a query and a virtual domain is an XML document.)(Page 617); and defining at least one virtual resource describing as least one actual resource within the at least one virtual domain to satisfy the requirement analysis (i.e. *"Continuing the example, suppose that a user wants to publish the purchase-order database as a list of orders in the XML format shown in Figure 3. There, each order appears as a top-level element, with its associated items and payments (ordered by due date) nested under it. To transform the default view into the desired XML format, a user-defined view called "orders" is created, as shown in Figure 4."* The preceding text clearly indicates that at least one virtual resource is the XML view of orders and one actual resource is the purchase-order database.)(Page 620).

As per claim 28, Funderburk teaches a system further comprising: analyzing a requirement for actual resource usage, to provide said requirement analysis (i.e. *"The XTABLES default XML view captures both relational data and meta-data (schema) information. This allows users to write queries (and create views) that treat relational data and meta-data interchangeably."* The preceding text clearly indicates that the actual resource usage is the relational data and the requirement analysis is the query.)(Page 618).

As per claim 29, Funderburk teaches a system further comprising: defining at least one virtual relationship between at least two virtual resources (i.e. *"To transform the default view into the desired XML format, a user-defined view called "orders" is created, as shown in Figure 4."* The preceding text clearly indicates that at least one virtual relationship, which is 'orders' exists between two virtual resources, which are the default XML view and the "orders" XML view.)(Page 620).

As per claim 30, Funderburk teaches a system wherein at least one of a virtual resource and a virtual relationship are utilized to create an application program (i.e. *"Once the 'orders' view has been created, queries can be issued against it."* The preceding text clearly indicates that an application program, which is the 'orders' view has been created, since queries can be issued against it, that is a set of instruction codes used to perform some tangible result.)(Page 621).

As per claim 33, Funderburk teaches a method of refactoring actual resources without alteration into a collection of virtual resources customized to a particular audience, comprising (i.e. *"As a starting point, XTABLES automatically creates the default XML view, which is a low-level XML view of the underlying relational database"* The preceding text clearly indicates that refactoring actual resources without alteration into a collection of virtual resource is the creation of a default XML view, where the XML view is the virtual resource and the database table is the actual resource.)(Page 620): providing a structured meta-data layer which contains semantic information for leveraging by a consumer of the virtual resources (i.e. *"Users can then create application-specific XML views on top of the default XML view."* The preceding text clearly indicates that a structured a consumer, which is the user, can leverage, which is create application-specific XML views, from the virtual resource, which is the default XML view.)(Page 616).

As per claim 34, the claim is rejected based on the dependency of claim 33.

As per claim 35, the claim is rejected based on the dependency of claim 33.

As per claim 36 and 37, Funderburk teaches a method further comprising:  
creating at least one virtual resource instance (i.e. "As a starting point, XTABLES automatically creates the default XML view, which is a low-level XML view of the underlying relational database." The preceding text clearly indicates that at least one virtual resource instance is created, which is the default XML view.)(Page 620); assigning an identity to the at least one virtual resource instance (i.e. *"To transform the default view into the desired XML format, a user-defined view called 'orders' is created, as shown in Figure 4."* The preceding text clearly indicates that an identity, 'orders' is assigned to the virtual resource instance, which is the user-defined XML view.)(Page 620); and associating the at least one virtual resource instance with one virtual resource (i.e. The term 'order' is assigned to the XML view.)(Page 620).

#### ***Contact Information***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Farhan M. Syed whose telephone number is 571-272-7191. The examiner can normally be reached on 8:30AM-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jeffrey Gaffin can be reached on 571-272-4146. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.



Art Unit: 2165

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

FMS



JEFFREY GAFFIN  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100